Package 'IncidencePrevalence'

March 8, 2025

Title Estimate Incidence and Prevalence using the OMOP Common Data Model

Version 1.2.0

Description Calculate incidence and prevalence using data mapped to the Observational Medical Outcomes Partnership (OMOP) common data model. Incidence and prevalence can be estimated for the total population in a database or for a stratification cohort.

Encoding UTF-8

RoxygenNote 7.3.2

Depends R (>= 4.1)

Imports CDMConnector (>= 2.0.0), cli (>= 3.0.0), clock, dplyr (>= 1.1.0), glue (>= 1.5.0), omopgenerics (>= 1.1.0), magrittr (>= 2.0.0), PatientProfiles (>= 1.3.1), purrr (>= 0.3.5), rlang (>= 1.0.0), stringr (>= 1.5.0), tidyr (>= 1.2.0)

Suggests knitr, dbplyr, rmarkdown, RPostgres, duckdb (>= 1.0.0), DBI (>= 1.0.0), odbc, here, Hmisc, epitools, tictoc, testthat (>= 0.3.1), spelling, gt, flextable, ggplot2 (>= 3.4.0), scales (>= 1.1.0), visOmopResults (>= 1.0.2), patchwork, binom

Config/testthat/edition 3

Config/testthat/parallel true

VignetteBuilder knitr

Language en-US

URL `https://darwin-eu.github.io/IncidencePrevalence/`

LazyData true

NeedsCompilation no

Author Edward Burn [aut, cre] (<`https://orcid.org/0000-0002-9286-1128`>),
Berta Raventos [aut] (<`https://orcid.org/0000-0002-4668-2970`>),
Marti Catala [aut] (<`https://orcid.org/0000-0003-3308-9905`>),
Mike Du [ctb] (<`https://orcid.org/0000-0002-9517-8834`>),
Yuchen Guo [ctb] (<`https://orcid.org/0000-0002-0847-4855`>),
Adam Black [ctb] (<`https://orcid.org/0000-0001-5576-8701`>),

Ger Inberg [ctb] (<https://orcid.org/0000-0001-8993-8748>),
Kim Lopez [ctb] (<https://orcid.org/0000-0002-8462-8668>)

Maintainer Edward Burn <edward.burn@ndorms.ox.ac.uk>

Repository CRAN

Date/Publication 2025-03-08 10:10:02 UTC

Contents

asIncidenceResult 2
asPrevalenceResult 3
availableIncidenceGrouping 4
availablePrevalenceGrouping 4
benchmarkIncidencePrevalence 5
estimateIncidence 6
estimatePeriodPrevalence 8
estimatePointPrevalence 9
generateDenominatorCohortSet 11
generateTargetDenominatorCohortSet 12
IncidencePrevalenceBenchmarkResults 14
mockIncidencePrevalence 14
optionsTableIncidence 16
optionsTablePrevalence 16
plotIncidence 17
plotIncidencePopulation 18
plotPrevalence 19
plotPrevalencePopulation 20
tableIncidence 21
tableIncidenceAttrition 22
tablePrevalence 23
tablePrevalenceAttrition 25

Index **26**

`asIncidenceResult` *A tidy implementation of the summarised_result object for incidence results.*

Description

A tidy implementation of the summarised_result object for incidence results.

Usage

```
asIncidenceResult(result, metadata = FALSE)
```

Arguments

`result`	A summarised_result object created by the IncidencePrevalence package.
`metadata`	If TRUE additional metadata columns will be included in the result.

Value

A tibble with a tidy version of the summarised_result object.

Examples

```
cdm <- mockIncidencePrevalence()
inc <- estimateIncidence(cdm, "target", "outcome")
tidy_inc <- asIncidenceResult(inc)
```

`asPrevalenceResult` *A tidy implementation of the summarised_result object for prevalence results.*

Description

A tidy implementation of the summarised_result object for prevalence results.

Usage

```
asPrevalenceResult(result, metadata = FALSE)
```

Arguments

`result`	A summarised_result object created by the IncidencePrevalence package.
`metadata`	If TRUE additional metadata columns will be included in the result.

Value

A tibble with a tidy version of the summarised_result object.

Examples

```
cdm <- mockIncidencePrevalence()
prev <- estimatePointPrevalence(cdm, "target", "outcome")
tidy_prev <- asPrevalenceResult(prev)
```

`availableIncidenceGrouping` *Variables that can be used for faceting and colouring incidence plots*

Description

Variables that can be used for faceting and colouring incidence plots

Usage

```
availableIncidenceGrouping(result, varying = FALSE)
```

Arguments

`result`	Incidence results
`varying`	If FALSE, only variables with non-unique values will be returned, otherwise all available variables will be returned

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 1000)
cdm <- generateDenominatorCohortSet(
  cdm = cdm, name = "denominator",
  cohortDateRange = c(as.Date("2014-01-01"), as.Date("2018-01-01"))
)
inc <- estimateIncidence(
  cdm = cdm,
  denominatorTable = "denominator",
  outcomeTable = "outcome"
)
availableIncidenceGrouping(inc)
```

`availablePrevalenceGrouping` *Variables that can be used for faceting and colouring prevalence plots*

Description

Variables that can be used for faceting and colouring prevalence plots

Usage

```
availablePrevalenceGrouping(result, varying = FALSE)
```

Arguments

`result`	Prevalence results
`varying`	If FALSE, only variables with non-unique values will be returned, otherwise all available variables will be returned

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 1000)
cdm <- generateDenominatorCohortSet(
  cdm = cdm, name = "denominator",
  cohortDateRange = c(as.Date("2014-01-01"), as.Date("2018-01-01"))
)
prev <- estimatePointPrevalence(
  cdm = cdm,
  denominatorTable = "denominator",
  outcomeTable = "outcome"
)
availablePrevalenceGrouping(prev)
```

`benchmarkIncidencePrevalence` *Run benchmark of incidence and prevalence analyses*

Description

Run benchmark of incidence and prevalence analyses

Usage

```
benchmarkIncidencePrevalence(cdm, analysisType = "all")
```

Arguments

`cdm`	A CDM reference object
`analysisType`	A string of the following: "all", "only incidence", "only prevalence"

Value

a tibble with time taken for different analyses

Examples

```
cdm <- mockIncidencePrevalence(
  sampleSize = 100,
  earliestObservationStartDate = as.Date("2010-01-01"),
  latestObservationStartDate = as.Date("2010-01-01"),
  minDaysToObservationEnd = 364,
  maxDaysToObservationEnd = 364,
  outPre = 0.1
)

timings <- benchmarkIncidencePrevalence(cdm)
```

`estimateIncidence` *Collect population incidence estimates*

Description

Collect population incidence estimates

Usage

```
estimateIncidence(
  cdm,
  denominatorTable,
  outcomeTable,
  censorTable = NULL,
  denominatorCohortId = NULL,
  outcomeCohortId = NULL,
  censorCohortId = NULL,
  interval = "years",
  completeDatabaseIntervals = TRUE,
  outcomeWashout = Inf,
  repeatedEvents = FALSE,
  strata = list(),
  includeOverallStrata = TRUE
)
```

Arguments

`cdm`	A CDM reference object
`denominatorTable`	A cohort table with a set of denominator cohorts (for example, created using the `generateDenominatorCohortSet()` function).
`outcomeTable`	A cohort table in the cdm reference containing a set of outcome cohorts.

`censorTable`	A cohort table in the cdm reference containing a cohort to be used for censoring. Individuals will stop contributing time at risk from the date of their first record in the censor cohort. If they appear in the censor cohort before entering the denominator cohort they will be excluded. The censor cohort can only contain one record per individual.
`denominatorCohortId`	The cohort definition ids or the cohort names of the denominator cohorts of interest. If NULL all cohorts will be considered in the analysis.
`outcomeCohortId`	The cohort definition ids or the cohort names of the outcome cohorts of interest. If NULL all cohorts will be considered in the analysis.
`censorCohortId`	The cohort definition id or the cohort name of the cohort to be used for censoring. Must be specified if there are multiple cohorts in the censor table.
`interval`	Time intervals over which incidence is estimated. Can be "weeks", "months", "quarters", "years", or "overall". ISO weeks will be used for weeks. Calendar months, quarters, or years can be used, or an overall estimate for the entire time period observed (from earliest cohort start to last cohort end) can also be estimated. If more than one option is chosen then results will be estimated for each chosen interval.
`completeDatabaseIntervals`	TRUE/ FALSE. Where TRUE, incidence will only be estimated for those intervals where the denominator cohort captures all the interval.
`outcomeWashout`	The number of days used for a 'washout' period between the end of one outcome and an individual starting to contribute time at risk. If Inf, no time can be contributed after an event has occurred.
`repeatedEvents`	TRUE/ FALSE. If TRUE, an individual will be able to contribute multiple events during the study period (time while they are present in an outcome cohort and any subsequent washout will be excluded). If FALSE, an individual will only contribute time up to their first event.
`strata`	Variables added to the denominator cohort table for which to stratify estimates.
`includeOverallStrata`	Whether to include an overall result as well as strata specific results (when strata has been specified).

Value

Incidence estimates

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 1000)
cdm <- generateDenominatorCohortSet(
  cdm = cdm, name = "denominator",
  cohortDateRange = c(as.Date("2008-01-01"), as.Date("2018-01-01"))
)
inc <- estimateIncidence(
  cdm = cdm,
```

```
  denominatorTable = "denominator",
  outcomeTable = "outcome"
)
```

`estimatePeriodPrevalence` *Estimate period prevalence*

Description

Estimate period prevalence

Usage

```
estimatePeriodPrevalence(
  cdm,
  denominatorTable,
  outcomeTable,
  denominatorCohortId = NULL,
  outcomeCohortId = NULL,
  interval = "years",
  completeDatabaseIntervals = TRUE,
  fullContribution = FALSE,
  strata = list(),
  includeOverallStrata = TRUE
)
```

Arguments

`cdm`	A CDM reference object
`denominatorTable`	A cohort table with a set of denominator cohorts (for example, created using the `generateDenominatorCohortSet()` function).
`outcomeTable`	A cohort table in the cdm reference containing a set of outcome cohorts.
`denominatorCohortId`	The cohort definition ids or the cohort names of the denominator cohorts of interest. If NULL all cohorts will be considered in the analysis.
`outcomeCohortId`	The cohort definition ids or the cohort names of the outcome cohorts of interest. If NULL all cohorts will be considered in the analysis.
`interval`	Time intervals over which period prevalence is estimated. This can be "weeks", "months", "quarters", "years", or "overall". ISO weeks will be used for weeks. Calendar months, quarters, or years can be used as the period. If more than one option is chosen then results will be estimated for each chosen interval.

`completeDatabaseIntervals`	TRUE/ FALSE. Where TRUE, prevalence will only be estimated for those intervals where the database captures all the interval (based on the earliest and latest observation period start dates, respectively).
`fullContribution`	TRUE/ FALSE. Where TRUE, individuals will only be included if they in the database for the entire interval of interest. If FALSE they are only required to present for one day of the interval in order to contribute.
`strata`	Variables added to the denominator cohort table for which to stratify estimates.
`includeOverallStrata`	Whether to include an overall result as well as strata specific results (when strata has been specified).

Value

Period prevalence estimates

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 1000)
cdm <- generateDenominatorCohortSet(
  cdm = cdm, name = "denominator",
  cohortDateRange = c(as.Date("2008-01-01"), as.Date("2018-01-01"))
)
estimatePeriodPrevalence(
  cdm = cdm,
  denominatorTable = "denominator",
  outcomeTable = "outcome",
  interval = "months"
)
```

`estimatePointPrevalence` *Estimate point prevalence*

Description

Estimate point prevalence

Usage

```
estimatePointPrevalence(
  cdm,
  denominatorTable,
  outcomeTable,
  denominatorCohortId = NULL,
  outcomeCohortId = NULL,
```

```
  interval = "years",
  timePoint = "start",
  strata = list(),
  includeOverallStrata = TRUE
)
```

Arguments

`cdm`	A CDM reference object
`denominatorTable`	A cohort table with a set of denominator cohorts (for example, created using the `generateDenominatorCohortSet()` function).
`outcomeTable`	A cohort table in the cdm reference containing a set of outcome cohorts.
`denominatorCohortId`	The cohort definition ids or the cohort names of the denominator cohorts of interest. If NULL all cohorts will be considered in the analysis.
`outcomeCohortId`	The cohort definition ids or the cohort names of the outcome cohorts of interest. If NULL all cohorts will be considered in the analysis.
`interval`	Time intervals over which period prevalence is estimated. Can be "weeks", "months", "quarters", or "years". ISO weeks will be used for weeks. Calendar months, quarters, or years can be used as the period. If more than one option is chosen then results will be estimated for each chosen interval.
`timePoint`	where to compute the point prevalence
`strata`	Variables added to the denominator cohort table for which to stratify estimates.
`includeOverallStrata`	Whether to include an overall result as well as strata specific results (when strata has been specified).

Value

Point prevalence estimates

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 1000)
cdm <- generateDenominatorCohortSet(
  cdm = cdm, name = "denominator",
  cohortDateRange = c(as.Date("2008-01-01"), as.Date("2018-01-01"))
)
estimatePointPrevalence(
  cdm = cdm,
  denominatorTable = "denominator",
  outcomeTable = "outcome",
  interval = "months"
)
```

`generateDenominatorCohortSet`

Identify a set of denominator populations

Description

`generateDenominatorCohortSet()` creates a set of cohorts that can be used for the denominator population in analyses of incidence, using `estimateIncidence()`, or prevalence, using `estimatePointPrevalence()` or `estimatePeriodPrevalence()`.

Usage

```
generateDenominatorCohortSet(
  cdm,
  name,
  cohortDateRange = as.Date(c(NA, NA)),
  ageGroup = list(c(0, 150)),
  sex = "Both",
  daysPriorObservation = 0,
  requirementInteractions = TRUE
)
```

Arguments

`cdm`	A CDM reference object
`name`	Name of the cohort table to be created. Note if a table already exists with this name in the database (give the prefix being used for the cdm reference) it will be overwritten.
`cohortDateRange`	Two dates. The first indicating the earliest cohort start date and the second indicating the latest possible cohort end date. If NULL or the first date is set as missing, the earliest observation_start_date in the observation_period table will be used for the former. If NULL or the second date is set as missing, the latest observation_end_date in the observation_period table will be used for the latter.
`ageGroup`	A list of age groups for which cohorts will be generated. A value of `list(c(0,17), c(18,30))` would, for example, lead to the creation of cohorts for those aged from 0 to 17, and from 18 to 30. In this example an individual turning 18 during the time period would appear in both cohorts (leaving the first cohort the day before their 18th birthday and entering the second from the day of their 18th birthday).
`sex`	Sex of the cohorts. This can be one or more of: `"Male"`, `"Female"`, or `"Both"`.
`daysPriorObservation`	The number of days of prior observation observed in the database required for an individual to start contributing time in a cohort.

`requirementInteractions`	If TRUE, cohorts will be created for all combinations of ageGroup, sex, and daysPriorObservation. If FALSE, only the first value specified for the other factors will be used. Consequently, order of values matters when requirementInteractions is FALSE.

Value

A cdm reference

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 1000)
cdm <- generateDenominatorCohortSet(
  cdm = cdm,
  name = "denominator",
  cohortDateRange = as.Date(c("2008-01-01", "2020-01-01"))
)
cdm
```

`generateTargetDenominatorCohortSet` *Identify a set of denominator populations using a target cohort*

Description

`generateTargetDenominatorCohortSet()` creates a set of cohorts that can be used for the denominator population in analyses of incidence, using `estimateIncidence()`, or prevalence, using `estimatePointPrevalence()` or `estimatePeriodPrevalence()`.

Usage

```
generateTargetDenominatorCohortSet(
  cdm,
  name,
  targetCohortTable,
  targetCohortId = NULL,
  cohortDateRange = as.Date(c(NA, NA)),
  timeAtRisk = c(0, Inf),
  ageGroup = list(c(0, 150)),
  sex = "Both",
  daysPriorObservation = 0,
  requirementsAtEntry = TRUE,
  requirementInteractions = TRUE
)
```

Arguments

`cdm`	A CDM reference object
`name`	Name of the cohort table to be created.
`targetCohortTable`	A cohort table in the cdm reference to use to limit cohort entry and exit (with individuals only contributing to a cohort when they are contributing to the cohort in the target table).
`targetCohortId`	The cohort definition ids or the cohort names of the cohorts of interest for the target table. If NULL all cohorts will be considered in the analysis.
`cohortDateRange`	Two dates. The first indicating the earliest cohort start date and the second indicating the latest possible cohort end date. If NULL or the first date is set as missing, the earliest observation_start_date in the observation_period table will be used for the former. If NULL or the second date is set as missing, the latest observation_end_date in the observation_period table will be used for the latter.
`timeAtRisk`	Lower and upper bound for the time at risk window to apply relative to the target cohort entry. A value of list(c(0, 30), c(31, 60)) would, for example, create one set of denominator cohorts with time up to the 30 days following target cohort entry and another set with time from 31 days following entry to 60 days. If time at risk start is after target cohort exit and/ or observation period end then no time will be contributed. If time at risk end is after cohort exit and/ or observation period, then only time up to these will be contributed.
`ageGroup`	A list of age groups for which cohorts will be generated. A value of `list(c(0,17), c(18,30))` would, for example, lead to the creation of cohorts for those aged from 0 to 17, and from 18 to 30.
`sex`	Sex of the cohorts. This can be one or more of: `"Male"`, `"Female"`, or `"Both"`.
`daysPriorObservation`	The number of days of prior observation observed in the database required for an individual to start contributing time in a cohort.
`requirementsAtEntry`	If TRUE, individuals must satisfy requirements for inclusion on their cohort start date for the target cohort. If FALSE, individuals will be included once they satisfy all requirements.
`requirementInteractions`	If TRUE, cohorts will be created for all combinations of ageGroup, sex, and daysPriorObservation. If FALSE, only the first value specified for the other factors will be used. Consequently, order of values matters when requirementInteractions is FALSE.

Value

A cdm reference

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 1000)
```

```
cdm <- generateTargetDenominatorCohortSet(
  cdm = cdm,
  name = "denominator",
  targetCohortTable = "target",
  cohortDateRange = as.Date(c("2008-01-01", "2020-01-01"))
)
cdm
```

`IncidencePrevalenceBenchmarkResults` *Benchmarking results*

Description

Benchmarking results

Usage

```
IncidencePrevalenceBenchmarkResults
```

Format

A list of results from benchmarking

`mockIncidencePrevalence` *Generate example subset of the OMOP CDM for estimating incidence and prevalence*

Description

Generate example subset of the OMOP CDM for estimating incidence and prevalence

Usage

```
mockIncidencePrevalence(
  personTable = NULL,
  observationPeriodTable = NULL,
  targetCohortTable = NULL,
  outcomeTable = NULL,
  censorTable = NULL,
  sampleSize = 1,
  outPre = 1,
  seed = 444,
  earliestDateOfBirth = NULL,
```

```
  latestDateOfBirth = NULL,
  earliestObservationStartDate = as.Date(”1900-01-01”),
  latestObservationStartDate = as.Date(”2010-01-01”),
  minDaysToObservationEnd = 1,
  maxDaysToObservationEnd = 4380,
  minOutcomeDays = 1,
  maxOutcomeDays = 10,
  maxOutcomes = 1
)
```

Arguments

`personTable`	A tibble in the format of the person table.
`observationPeriodTable`	A tibble in the format of the observation period table.
`targetCohortTable`	A tibble in the format of a cohort table which can be used for stratification
`outcomeTable`	A tibble in the format of a cohort table which can be used for outcomes
`censorTable`	A tibble in the format of a cohort table which can be used for censoring
`sampleSize`	The number of unique patients.
`outPre`	The fraction of patients with an event.
`seed`	The seed for simulating the data set. Use the same seed to get same data set.
`earliestDateOfBirth`	The earliest date of birth of a patient in person table.
`latestDateOfBirth`	The latest date of birth of a patient in person table.
`earliestObservationStartDate`	The earliest observation start date for patient format.
`latestObservationStartDate`	The latest observation start date for patient format.
`minDaysToObservationEnd`	The minimum number of days of the observational integer.
`maxDaysToObservationEnd`	The maximum number of days of the observation period integer.
`minOutcomeDays`	The minimum number of days of the outcome period default set to 1.
`maxOutcomeDays`	The maximum number of days of the outcome period default set to 10.
`maxOutcomes`	The maximum possible number of outcomes per person can have default set to 1.

Value

A cdm reference to a duckdb database with mock data.

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 100)
cdm
```

`optionsTableIncidence` *Additional arguments for the functions tableIncidence.*

Description

It provides a list of allowed inputs for .option argument in tableIncidence, and their given default values.

Usage

```
optionsTableIncidence()
```

Value

The default .options named list.

Examples

```
{
  optionsTableIncidence()
}
```

`optionsTablePrevalence` *Additional arguments for the functions tablePrevalence.*

Description

It provides a list of allowed inputs for .option argument in tablePrevalence, and their given default values.

Usage

```
optionsTablePrevalence()
```

Value

The default .options named list.

Examples

```
{
  optionsTablePrevalence()
}
```

`plotIncidence` *Plot incidence results*

Description

Plot incidence results

Usage

```
plotIncidence(
  result,
  x = "incidence_start_date",
  y = "incidence_100000_pys",
  line = FALSE,
  point = TRUE,
  ribbon = FALSE,
  ymin = "incidence_100000_pys_95CI_lower",
  ymax = "incidence_100000_pys_95CI_upper",
  facet = NULL,
  colour = NULL
)
```

Arguments

`result`	Incidence results
`x`	Variable to plot on x axis
`y`	Variable to plot on y axis.
`line`	Whether to plot a line using `geom_line`
`point`	Whether to plot points using `geom_point`
`ribbon`	Whether to plot a ribbon using `geom_ribbon`
`ymin`	Lower limit of error bars, if provided is plot using `geom_errorbar`
`ymax`	Upper limit of error bars, if provided is plot using `geom_errorbar`
`facet`	Variables to use for facets. To see available variables for facetting use the function `availableIncidenceGrouping()`.
`colour`	Variables to use for colours. To see available variables for colouring use the function `availableIncidenceGrouping()`.

Value

A ggplot with the incidence results plotted

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 1000)
cdm <- generateDenominatorCohortSet(
  cdm = cdm, name = "denominator",
  cohortDateRange = c(as.Date("2008-01-01"), as.Date("2018-01-01"))
)
inc <- estimateIncidence(
  cdm = cdm,
  denominatorTable = "denominator",
  outcomeTable = "outcome"
)
plotIncidence(inc)
```

`plotIncidencePopulation` *Bar plot of denominator counts, outcome counts, and person-time from incidence results*

Description

Bar plot of denominator counts, outcome counts, and person-time from incidence results

Usage

```
plotIncidencePopulation(
  result,
  x = "incidence_start_date",
  y = "denominator_count",
  facet = NULL,
  colour = NULL
)
```

Arguments

`result`	Incidence results
`x`	Variable to plot on x axis
`y`	Variable to plot on y axis.
`facet`	Variables to use for facets. To see available variables for facetting use the functions `availableIncidenceGrouping()`.
`colour`	Variables to use for colours. To see available variables for colouring use the function `availableIncidenceGrouping()`.

Value

A ggplot object

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 1000)
cdm <- generateDenominatorCohortSet(
  cdm = cdm, name = "denominator",
  cohortDateRange = c(as.Date("2014-01-01"), as.Date("2018-01-01"))
)
inc <- estimateIncidence(
  cdm = cdm,
  denominatorTable = "denominator",
  outcomeTable = "outcome"
)
plotIncidencePopulation(inc)
```

`plotPrevalence` *Plot prevalence results*

Description

Plot prevalence results

Usage

```
plotPrevalence(
  result,
  x = "prevalence_start_date",
  y = "prevalence",
  line = FALSE,
  point = TRUE,
  ribbon = FALSE,
  ymin = "prevalence_95CI_lower",
  ymax = "prevalence_95CI_upper",
  facet = NULL,
  colour = NULL
)
```

Arguments

`result`	Prevalence results
`x`	Variable to plot on x axis
`y`	Variable to plot on y axis.
`line`	Whether to plot a line using `geom_line`

`point`	Whether to plot points using `geom_point`
`ribbon`	Whether to plot a ribbon using `geom_ribbon`
`ymin`	Lower limit of error bars, if provided is plot using `geom_errorbar`
`ymax`	Upper limit of error bars, if provided is plot using `geom_errorbar`
`facet`	Variables to use for facets. To see available variables for facetting use the function `availablePrevalenceGrouping()`.
`colour`	Variables to use for colours. To see available variables for colouring use the function `availablePrevalenceGrouping()`.

Value

A ggplot with the prevalence results plotted

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 1000)
cdm <- generateDenominatorCohortSet(
  cdm = cdm, name = "denominator",
  cohortDateRange = c(as.Date("2014-01-01"), as.Date("2018-01-01"))
)
prev <- estimatePointPrevalence(
  cdm = cdm,
  denominatorTable = "denominator",
  outcomeTable = "outcome"
)
plotPrevalence(prev)
```

`plotPrevalencePopulation`

Bar plot of denominator and outcome counts from prevalence results

Description

Bar plot of denominator and outcome counts from prevalence results

Usage

```
plotPrevalencePopulation(
  result,
  x = "prevalence_start_date",
  y = "denominator_count",
  facet = NULL,
  colour = NULL
)
```

Arguments

`result`	Prevalence results
`x`	Variable to plot on x axis
`y`	Variable to plot on y axis.
`facet`	Variables to use for facets. To see available variables for facetting use the functions `availablePrevalenceGrouping()`.
`colour`	Variables to use for colours. To see available variables for colouring use the function `availablePrevalenceGrouping()`.

Value

A ggplot object

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 1000)
cdm <- generateDenominatorCohortSet(
  cdm = cdm, name = "denominator",
  cohortDateRange = c(as.Date("2014-01-01"), as.Date("2018-01-01"))
)
prev <- estimatePointPrevalence(
  cdm = cdm,
  denominatorTable = "denominator",
  outcomeTable = "outcome"
)
plotPrevalencePopulation(prev)
```

`tableIncidence` *Table of incidence results*

Description

Table of incidence results

Usage

```
tableIncidence(
  result,
  type = "gt",
  header = c("estimate_name"),
  groupColumn = c("cdm_name", "outcome_cohort_name"),
  settingsColumn = c("denominator_age_group", "denominator_sex"),
  hide = c("denominator_cohort_name", "analysis_interval"),
  .options = list()
)
```

Arguments

`result`	Incidence results
`type`	Type of table. Can be "gt", "flextable", or "tibble"
`header`	A vector specifying the elements to include in the header. The order of elements matters, with the first being the topmost header. The header vector can contain one of the following variables: "cdm_name", "denominator_cohort_name", "outcome_cohort_name", "incidence_start_date", "incidence_end_date", "estimate_name", variables in the `strata_name` column, and any of the settings columns specified in `settingsColumn` argument. The header can also include other names to use as overall header labels
`groupColumn`	Variables to use as group labels. Allowed columns are the same as in `header`
`settingsColumn`	Variables from the settings attribute to display in the table
`hide`	Table columns to exclude, options are the ones described in `header`
`.options`	Table options to apply

Value

Table of results

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 1000)
cdm <- generateDenominatorCohortSet(
  cdm = cdm, name = "denominator",
  cohortDateRange = c(as.Date("2008-01-01"), as.Date("2018-01-01"))
)
inc <- estimateIncidence(
  cdm = cdm,
  denominatorTable = "denominator",
  outcomeTable = "outcome"
)
tableIncidence(inc)
```

`tableIncidenceAttrition`

Table of incidence attrition results

Description

Table of incidence attrition results

Usage

```
tableIncidenceAttrition(
  result,
  type = "gt",
  header = "variable_name",
  groupColumn = c("cdm_name", "outcome_cohort_name"),
  settingsColumn = NULL,
 hide = c("denominator_cohort_name", "estimate_name", "reason_id", "variable_level")
)
```

Arguments

`result`	A summarised_result object. Output of summariseCohortAttrition().
`type`	Type of table. Check supported types with `visOmopResults::tableType()`.
`header`	Columns to use as header. See options with `colnames(omopgenerics::splitAll(result))`. Variables in `settingsColumn` are also allowed
`groupColumn`	Variables to use as group labels. Allowed columns are the same as in `header`
`settingsColumn`	Variables from the settings attribute to display in the table
`hide`	Table columns to exclude, options are the ones described in `header`

Value

A visual table.

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 1000)
cdm <- generateDenominatorCohortSet(
  cdm = cdm, name = "denominator",
  cohortDateRange = c(as.Date("2008-01-01"), as.Date("2018-01-01"))
)
inc <- estimateIncidence(
  cdm = cdm,
  denominatorTable = "denominator",
  outcomeTable = "outcome"
)
tableIncidenceAttrition(inc)
```

`tablePrevalence` *Table of prevalence results*

Description

Table of prevalence results

Usage

```
tablePrevalence(
  result,
  type = "gt",
  header = c("estimate_name"),
  groupColumn = c("cdm_name", "outcome_cohort_name"),
  settingsColumn = c("denominator_age_group", "denominator_sex"),
  hide = c("denominator_cohort_name", "analysis_interval"),
  .options = list()
)
```

Arguments

`result`	Prevalence results
`type`	Type of table. Can be "gt", "flextable", or "tibble"
`header`	A vector specifying the elements to include in the header. The order of elements matters, with the first being the topmost header. The header vector can contain one of the following variables: "cdm_name", "denominator_cohort_name", "outcome_cohort_name", "prevalence_start_date", "prevalence_end_date", "estimate_name", variables in the `strata_name` column, and any of the settings columns specified in `settingsColumn` argument. The header can also include other names to use as overall header labels
`groupColumn`	Variables to use as group labels. Allowed columns are the same as in `header`
`settingsColumn`	Variables from the settings attribute to display in the table
`hide`	Table columns to exclude, options are the ones described in `header`
`.options`	Table options to apply

Value

Table of prevalence results

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 1000)
cdm <- generateDenominatorCohortSet(
  cdm = cdm,
  name = "denominator",
  cohortDateRange = c(as.Date("2008-01-01"), as.Date("2018-01-01"))
)
prev <- estimatePointPrevalence(
  cdm = cdm,
  denominatorTable = "denominator",
  outcomeTable = "outcome",
  interval = "months"
)
tablePrevalence(prev)
```

`tablePrevalenceAttrition` *Table of prevalence attrition results*

Description

Table of prevalence attrition results

Usage

```
tablePrevalenceAttrition(
  result,
  type = "gt",
  header = "variable_name",
  groupColumn = c("cdm_name", "outcome_cohort_name"),
  settingsColumn = NULL,
 hide = c("denominator_cohort_name", "estimate_name", "reason_id", "variable_level")
)
```

Arguments

`result`	A summarised_result object. Output of summariseCohortAttrition().
`type`	Type of table. Check supported types with `visOmopResults::tableType()`.
`header`	Columns to use as header. See options with `colnames(omopgenerics::splitAll(result))`. Variables in `settingsColumn` are also allowed
`groupColumn`	Variables to use as group labels. Allowed columns are the same as in `header`
`settingsColumn`	Variables from the settings attribute to display in the table
`hide`	Table columns to exclude, options are the ones described in `header`

Value

A visual table.

Examples

```
cdm <- mockIncidencePrevalence(sampleSize = 1000)
cdm <- generateDenominatorCohortSet(
  cdm = cdm, name = "denominator",
  cohortDateRange = c(as.Date("2008-01-01"), as.Date("2018-01-01"))
)
prev <- estimatePointPrevalence(
  cdm = cdm,
  denominatorTable = "denominator",
  outcomeTable = "outcome",
  interval = "months"
)
tablePrevalenceAttrition(prev)
```

Index

∗ **datasets**
  IncidencePrevalenceBenchmarkResults, 14

asIncidenceResult, 2
asPrevalenceResult, 3
availableIncidenceGrouping, 4
availablePrevalenceGrouping, 4

benchmarkIncidencePrevalence, 5

estimateIncidence, 6
estimatePeriodPrevalence, 8
estimatePointPrevalence, 9

generateDenominatorCohortSet, 11
generateTargetDenominatorCohortSet, 12

IncidencePrevalenceBenchmarkResults, 14

mockIncidencePrevalence, 14

optionsTableIncidence, 16
optionsTablePrevalence, 16

plotIncidence, 17
plotIncidencePopulation, 18
plotPrevalence, 19
plotPrevalencePopulation, 20

tableIncidence, 21
tableIncidenceAttrition, 22
tablePrevalence, 23
tablePrevalenceAttrition, 25